**SQL Server - To create 2 tables with PK-FK relationship and insert and delete rows through Python Code**

import pyodbc

conn = pyodbc.connect('Driver={SQL Server};'

'Server=DESKTOP-KU0CRG6\SQLEXPRESS;'

'Database=hospital;'

'Trusted\_Connection=yes;')

cur = conn.cursor()

#The above statements are common for all sections

#sections are separated by the # lines

###########################

**#to create a Primary table doctor**

##cur.execute(''' create table doctor (Id INT PRIMARY KEY IDENTITY(1,1),

## Name VARCHAR (50) NOT NULL,

## Specialised VARCHAR(50) NOT NULL) ''')

##conn.commit()

##conn.close()

##########################

**#to create a Foreign table patient which has a FK DoctorId and refers to the PK Id in doctor table**

##cur.execute(''' create table patient ( Id INT PRIMARY KEY IDENTITY(1,1),

## Name VARCHAR (50) NOT NULL,

## DoctorId INT foreign key references doctor(Id)) ''')

##conn.commit()

##conn.close()

##############################

**#to insert data rows into doctor table**

##cur.execute(''' insert into doctor (Name,Specialised) values ('KrishnaKumar','Gynaecology'),

## ('Bhattacharya','General Physician') ,

('Anil Venkatachalam','Neurologist') ''')

##cur.execute('select \* from doctor')

##

##for row in cur:

## print(row)

##conn.commit()

##conn.close()

###########################

**#to insert rows into patient table**

##cur.execute(''' insert into patient (Name,DoctorId) values ('Rajneet',1),

## ('Sameer',2),

## ('Ratna',2),

## ('Sonu',3) ''')

##cur.execute('select \* from patient')

##

##for row in cur:

## print(row)

##conn.commit()

##conn.close()

**Updating Rows of the 2 Tables**

If there's a **Primary Key–Foreign Key (PK–FK)** relationship, **updating is allowed** but with conditions.

You **can update a table involved in a PK–FK relationship** **as long as** the **referential integrity is preserved**.

**Updating the PK (Primary Key) value**

* **NOT allowed** by default **if the FK table has dependent rows**.
* SQL Server will block the update to prevent breaking the relationship.

**Updating other columns in PK table (non-key columns)**

* **Allowed**

**Updating the FK value in the Foreign or child table**

* **Allowed**, but **only if the new FK value exists** in the parent table.

############################

"Due to the primary key–foreign key relationship, **any dependent rows** in the foreign key table must be deleted before deleting the corresponding row in the primary key table."

**#delete rows from patient table**

##cur.execute(''' delete from patient WHERE DoctorId= 2 ''')

##cur.execute('select \* from patient')

##

##for row in cur:

## print(row)

##conn.commit()

##conn.close()

########################

**#delete rows from doctor table**

##cur.execute(''' delete from doctor WHERE Id= 2 ''')

##cur.execute('select \* from doctor')

##

##for row in cur:

## print(row)

##conn.commit()

##conn.close()

###################